



# Dantzig-Wolfe and Lagrangian decompositions in integer linear programming

Lucas Létocart, Nora Touati MOUNGLA, Anass Nagih

## ► To cite this version:

Lucas Létocart, Nora Touati MOUNGLA, Anass Nagih. Dantzig-Wolfe and Lagrangian decompositions in integer linear programming. *International Journal of Mathematics in Operational Research*, 2011, 15 p. hal-00595996

**HAL Id: hal-00595996**

**<https://hal.science/hal-00595996>**

Submitted on 26 May 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Dantzig-Wolfe and Lagrangian decompositions in integer linear programming

L. Létocart<sup>a</sup> N. Touati-Moungla<sup>b</sup> A. Nagih<sup>c</sup>

<sup>a</sup>*LIPN, UMR 7030 CNRS, Institut Galilée - Université Paris 13, 99 avenue  
Jean-Baptiste Clément 93430 Villetaneuse, France.*

<sup>b</sup>*LIX, École polytechnique, 91128 Palaiseau Cedex, France.*

<sup>c</sup>*LITA, Université Paul Verlaine, Ile du Saulcy 57045 Metz Cedex 1, France.*

---

## Abstract

We propose in this paper a new Dantzig-Wolfe master model based on Lagrangian decomposition. We establish the relationship with classical Dantzig-Wolfe decomposition master problem and propose an alternative proof of the dominance of Lagrangian decomposition on Lagrangian relaxation dual bound. As illustration, we give the corresponding models and numerical results for two standard mathematical programs: the 0 – 1 bidimensional knapsack problem and the generalized assignment problem.

*Key words:* Dantzig-Wolfe decomposition, Column generation, Lagrangian decomposition, Lagrangian relaxation, 0-1 bidimensional knapsack problem, generalized assignment problem.

---

---

*Email addresses:* `lucas.letocart@lipn.univ-paris13.fr` (L. Létocart),  
`touati@lix.polytechnique.fr` (N. Touati-Moungla),  
`anass.nagih@univ-metz.fr` (A. Nagih).

## 1 Introduction

An integer linear program where constraints are partitioned in two subsets can be formulated as follows:

$$(P) \left\{ \begin{array}{l} \max \quad c^t x \\ s.c. \quad Ax = a \\ \quad \quad Bx = b \\ \quad \quad x \in X \end{array} \right.$$

where  $c \in \mathbb{R}^n$ ,  $A$  is a  $m \times n$  matrix,  $B$  is a  $p \times n$  matrix,  $a \in \mathbb{R}^m$ ,  $b \in \mathbb{R}^p$  and  $X \subseteq \mathbb{N}^n$ .

These problems are generally NP-hard and bounds are needed to solve them in generic branch and bound like schemes. To improve the bound based on the continuous relaxation of  $(P)$ , Lagrangian methods, like Lagrangian Relaxation (LR) [2], Lagrangian Decomposition (LD) [5,6,11,13,12], Lagrangian substitution [14] and Surrogate Relaxation (SR) [4], are well-known techniques for obtaining bounds in Integer Linear Programming (ILP).

This work recalls the existing link between LR and classical Dantzig-Wolfe Decomposition (DWD) [1] and establishes the relationship between LD and DWD to derive a new DW master model.

The equivalence between DWD and LR is well known [9]. Solving a linear program by Column Generation (CG), using DWD, is the same as solving the Lagrangian dual by Kelley's cutting plane method [7]. This work recalls the previous result and extends it to LD, which can be viewed as a specific DWD, to prove the superiority of the new bound obtained.

The paper is organized as follows. Section 2 deals with LR, LD and DWD principles. Section 3 shows the relationship between LD and DWD, and gives a new proof on the LD bound dominance over the LR one. In Section 4 we illustrate with two DW master models on the 0-1 Bi-dimensional Knapsack Problem (0-1.BKP) and the Generalized Assignment Problem (GAP). In section 5 we present some computational results on the two previous problems.

## 2 Lagrangian duals and Dantzig-Wolfe decomposition

These approaches can be used in the pre-treatment phase of an exact or heuristic method in order to compute better bounds than linear relaxation. In this

section, we recall the principle of Lagrangian duality and its link with DWD and Column Generation (CG).

### 2.1 Dual Lagrangian relaxation

LR consists in omitting some complicating constraints ( $Ax = a$ ) and in incorporating them in the objective function using a Lagrangian multiplier  $\pi \in \mathbb{R}^m$ . We obtain the following relaxation:

$$(LR(\pi)) \begin{cases} \max c^t x + \pi^t(a - Ax) \\ \text{s. c. } Bx = b \\ x \in X. \end{cases}$$

For any  $\pi \in \mathbb{R}^m$ , the value of  $(LR(\pi))$  is an upper bound on  $v(P)$ . The best one is given by the LR dual:

$$\begin{aligned} (LRD) &\equiv \min_{\pi \in \mathbb{R}^m} (LR(\pi)) \\ &\equiv \min_{\pi \in \mathbb{R}^m} \max_{\{x \in X, Bx=b\}} c^t x + \pi^t(a - Ax). \end{aligned}$$

Let be  $X_B = \{x \in X | Bx = b\}$  and  $Conv(X_B)$  its convex hull (boundary of the convex polygon), supposed bounded. We denoted by  $x^{(k)}, k \in \{1, \dots, K\}$  the extreme points of  $Conv(X_B)$ . Hence,  $(LRD)$  can be reformulated as follows:

$$\begin{aligned} (LRD) &\equiv \min_{\pi \in \mathbb{R}^m} \max_{k=1, \dots, K} c^t x^{(k)} + \pi^t(a - Ax^{(k)}) \\ &\equiv \begin{cases} \min z \\ \text{s.t. } z + \pi^t(Ax^{(k)} - a) \geq c^t x^{(k)}, \quad k = 1, \dots, K \\ \pi \in \mathbb{R}^m, z \in \mathbb{R}. \end{cases} \end{aligned}$$

This new formulation potentially contains an exponential number of constraints, equal to  $K$ . Kelley's cutting plans method [7] considers a reduced set of these constraints that handle a restricted problem. Cuts (constraints) are added at each iteration until the optimum reached.

## 2.2 Lagrangian decomposition dual

It is well-known that the efficiency of branch and bound like scheme depends on the quality of the bounds. To improve those provided by LR, Guignard and Kim [5,6] have proposed to use LD. In such an approach, copy constraints are added to the formulation  $(P)$  to build an equivalent problem:

$$\left\{ \begin{array}{l} \max \quad c^t x \\ \text{s. c.} \quad Ax = a \\ \quad \quad By = b \\ \quad \quad x = y \\ \quad \quad x \in X, \ y \in Y, \text{ with } Y \supseteq X \end{array} \right.$$

where the copy variables permits to split the initial problem in two independent sub-problems after applying LR on the copy constraints  $x = y$ :

$$(LD(w)) \quad \left\{ \begin{array}{l} \max \quad c^t x + w^t(y - x) \\ \text{s. c.} \quad Ay = a \\ \quad \quad Bx = b \\ \quad \quad x \in X, \ y \in Y, \end{array} \right.$$

where  $w \in \mathbb{R}^n$  are dual variables associated to the copy constraints. We obtain the two following independent sub-problems:

$$(LD_y(w)) \quad \left\{ \begin{array}{l} \max \quad w^t y \\ \text{s. c.} \quad Ay = a \\ \quad \quad y \in Y \end{array} \right. \quad \text{and} \quad (LD_x(w)) \quad \left\{ \begin{array}{l} \max \quad (c - w)^t x \\ \text{s. c.} \quad Bx = b \\ \quad \quad x \in X \end{array} \right.$$

The LD dual is given by

$$(LDD) \min_{w \in \mathbb{R}^n} v(LD(w))$$

where

$$v(LD(w)) = \max\{w^t y | y \in Y_A\} + \max\{(c - w)^t x | x \in X_B\}$$

with

$$Y_A = \{y \mid Ay = a, \ y \in Y\} \quad X_B = \{x \mid Bx = b, \ x \in X\}.$$

This dual can be rewritten as :

$$(LDD) \begin{cases} \min & \max(c - w)^t x + \max w^t y \\ w \in \mathbb{R}^n & x \in X_B \quad y \in Y_A. \end{cases}$$

We assume that the convex hull of the sets  $Y_A$  and  $X_B$  are bounded. We denote by  $x^{(k)}, k \in \{1, \dots, K\}$  the extreme points of  $X_B$  and by  $y^{(l)}, l \in \{1, \dots, L\}$  those of  $Y_A$ . We obtain the following formulation:

$$(LDD) \begin{cases} \min & \max(c - w)^t x^{(k)} + \max w^t y^{(l)} \\ w \in \mathbb{R}^n & k = 1, \dots, K \quad l = 1, \dots, L \end{cases}$$

which can be expressed in this equivalent linear form:

$$(LDD) \begin{cases} \min & z_1 + z_2 \\ & z_1 \geq (c - w)^t x^{(k)}, k = 1, \dots, K \\ & z_2 \geq w^t y^{(l)}, l = 1, \dots, L \\ & w \in \mathbb{R}^n, z_1, z_2 \in \mathbb{R}. \end{cases}$$

The following theorem give the well-known dominance relationship between  $(P)$ ,  $(LRD)$ ,  $(LDD)$  and  $(LP)$  which is the linear relaxation of  $(P)$ .

**Theorem 1** [5,6]  $v(P) \leq v(LDD) \leq v(LRD) \leq v(LP)$ .

### 2.3 Dantzig-Wolfe decomposition and column generation

The key idea of DWD [1] is to reformulate the problem by substituting the original variables with a convex combination of the extreme points of the polyhedron corresponding to a substructure of the formulation.

We know that

$$\forall x \in \text{Conv}(X_B), \quad x = \sum_{k=1}^K \lambda_k x^{(k)}$$

with  $\sum_{k=1}^K \lambda_k = 1, \lambda_k \geq 0, \forall k \in 1, \dots, K$ .

By substituting in  $(P)$  we obtain the master problem of DWD:

$$(MP) \begin{cases} \max \sum_{k=1}^K (c^t x^{(k)}) \lambda_k \\ \text{s.c.} \quad \sum_{k=1}^K (Ax^{(k)}) \lambda_k = a \\ \sum_{k=1}^K \lambda_k = 1 \\ \lambda_k \geq 0, \quad k = 1, \dots, K. \end{cases}$$

$(MP)$  contains  $m+1$  constraints and (potentially) a huge number of variables (i.e. the number  $K$  of extreme points of  $\text{Conv}(X_B)$ ).

**Remark 1** Due to the fact that  $(LRD)$  is a dual of  $(MP)$ ,  $v(LRD) = v(MP)$  [9].

CG consists in generating iteratively a subset of the extreme points of  $X_B$  to determine an optimal solution of  $(MP)$  by solving alternatively:

- a *Restricted Master Problem* of DWD on a subset  $\mathcal{K}$  of  $\{1, \dots, K\}$ :

$$(RMP) \begin{cases} \max \sum_{k \in \mathcal{K}} (c^t x^{(k)}) \lambda_k \\ \text{s.c.} \quad \sum_{k \in \mathcal{K}} (Ax^{(k)}) \lambda_k = a \\ \sum_{k \in \mathcal{K}} \lambda_k = 1 \\ \lambda_k \geq 0, \quad k \in \mathcal{K} \end{cases}$$

- a pricing problem:

$$(SP) \begin{cases} \max \quad c^t x - \pi^t Ax - \pi_0 \\ \text{s. c.} \quad Bx = b \\ x \in X \end{cases}$$

where  $(\pi, \pi_0) \in \mathbb{R}^m \times \mathbb{R}$  are the dual variables provided by the resolution of  $(RMP)$ . The solution of  $(SP)$  is incorporated (as a column) in  $(RMP)$  if its value is non negative.

This process ends when there is no more variables in  $\{1, \dots, K\} \setminus \mathcal{K}$  with a positive reduced cost.

### 3 Lagrangian and Dantzig-Wolfe decompositions

This section is dedicated to Lagrangian decomposition duality. We establish the relationship between LD, DWD and CG. We consider the following DW master problem :

$$(MPD) \begin{cases} \max \sum_{k=1}^K (cx^{(k)})\lambda_k \\ \sum_{k=1}^K x^{(k)}\lambda_k - \sum_{l=1}^L y^{(l)}\gamma_l = 0 \\ \sum_{k=1}^K \lambda_k = 1 \\ \sum_{l=1}^L \gamma_l = 1 \\ \lambda_k \geq 0, k = 1, \dots, K, \gamma_l \geq 0, l = 1, \dots, L \end{cases}$$

where  $x^{(k)}, k \in \{1, \dots, K\}$  the extreme points of  $X_B$  and by  $y^{(l)}, l \in \{1, \dots, L\}$  those of  $Y_A$ .

**Lemma 1** *The value of this master problem (MPD) provides a better upper bound on  $v(P)$  than the value of the classical DWD (MP).*

**Proof:**

$$v(MPD) = \begin{cases} \max \sum_{k=1}^K (cx^{(k)})\lambda_k \\ \sum_{k=1}^K x^{(k)}\lambda_k - \sum_{l=1}^L y^{(l)}\gamma_l = 0 \\ \sum_{k=1}^K \lambda_k = 1 \\ \sum_{l=1}^L \gamma_l = 1 \\ \lambda_k \geq 0, k = 1, \dots, K, \gamma_l \geq 0, l = 1, \dots, L. \end{cases}$$

By duality

$$v(MPD) = \begin{cases} \min z_1 + z_2 \\ z_1 + w^t x^{(k)} \geq cx^k, k = 1, \dots, K \quad (1) \\ z_2 - w^t y^{(l)} \geq 0, l = 1, \dots, L \quad (2) \\ w \in \mathbb{R}^n, z_1, z_2 \in \mathbb{R} \end{cases}$$

If we consider only a subset of the multipliers  $w \in \mathbb{R}^n$  such that  $w^t = \pi^t A$ , where  $\pi$  is a vector of  $\mathbb{R}^m$ , and substitute in (1) and (2) we obtain the following problem:

$$\begin{cases} \min z_1 + z_2 \\ z_1 + \pi^t A x^{(k)} \geq cx^k, k = 1, \dots, K \\ z_2 - \pi^t A y^{(l)} \geq 0, l = 1, \dots, L \\ w \in \mathbb{R}^n, z_1, z_2 \in \mathbb{R} \end{cases}$$



for which the dual is:

$$\left\{ \begin{array}{l} \max \sum_{k=1}^K (cx^{(k)})\lambda_k \\ \sum_{k=1}^K Ax^{(k)}\lambda_k - \sum_{l=1}^L Ay^{(l)}\gamma_l = 0 \\ \sum_{k=1}^K \lambda_k = 1 \\ \sum_{l=1}^L \gamma_l = 1 \\ \lambda_k \geq 0, k = 1, \dots, K, \gamma_l \geq 0, l = 1, \dots, L. \end{array} \right.$$

As  $y^{(l)}, l \in \{1, \dots, L\}$  are the extreme points of  $Y_A$ , we have  $Ay^{(l)} = a$ , and by  $\sum_l \gamma_l = 1$ , we obtain the problem (MPD). Thus  $v(MPD) \leq v(MP)$  ■

**Remark 2** If  $n > m$ , the set  $\{\pi^t A, \pi \in \mathbb{R}^m\} \subsetneq \mathbb{R}^n$  and then  $v(MPD)$  can be stricly better than  $v(MP)$ .

**Remark 3** As (LDD) (resp. (LRD)) is the dual of (MPD) (resp. (MP)), we can state that

$$v(MPD) = v(LDD) = \min_{w \in \mathbb{R}^n} v(LD(w)) \leq \min_{\pi^t \in \mathbb{R}^m} v(LD(\pi^t A))$$

and

$$\min_{\pi^t \in \mathbb{R}^m} v(LD(\pi^t A)) = \min_{\pi \in \mathbb{R}^m} v(LR(\pi)) = v(LRD) = v(MP).$$

This approach supply an alternative proof to the dominance of LD over LR.

## 4 Decomposition models

This section is devoted to an illustration of this new DWD model on two classical combinatorial optimization problems : the 0-1 bi-dimensional knapsack problem and the generalized assignment problem.

### 4.1 The 0-1 bi-dimensional knapsack problem

This problem consists in selecting a subset of given objects (or items) in such a way that the total profit of the selected objects is maximized while two knapsack constraints are satisfied. The formulation of this problem is given by :

$$(0-1\_BKP) \left\{ \begin{array}{l} \max \sum_{i=1}^n c_i x_i \\ \text{s. c. } \sum_{i=1}^n a_i x_i \leq A \\ \sum_{i=1}^n b_i x_i \leq B \\ x_i \in \{0, 1\}, \quad i = 1, \dots, n \end{array} \right.$$

where  $n$  is the number of objects (or items), the coefficients  $a_i$  ( $i = 1, \dots, n$ ),  $b_i$  ( $i = 1, \dots, n$ ) and  $c_i$  ( $i = 1, \dots, n$ ) are positive integers and  $A$  and  $B$  are integers such that  $\max\{a_i : i = 1, \dots, n\} \leq A < \sum_{i=1, \dots, n} a_i$  and  $\max\{b_i : i = 1, \dots, n\} \leq B < \sum_{i=1, \dots, n} b_i$ .

The classical Dantzig-Wolfe master problem is given by:

$$\left\{ \begin{array}{l} \max \sum_{k=1}^K (\sum_{i=1}^n c_i x_i^{(k)}) \lambda_k \\ \text{s.c. } \sum_{k=1}^K (\sum_{i=1}^n a_i x_i^{(k)}) \lambda_k \leq A \\ \sum_{k=1}^K \lambda_k = 1 \\ \lambda_k \geq 0, \quad k = 1, \dots, K. \end{array} \right.$$

where  $x^{(k)}, k = 1, \dots, K$ , are the extreme points of  $\text{Conv}(\{x_i \in \{0, 1\} \mid \sum_{i=1}^n b_i x_i \leq B, \quad i = 1, \dots, n\})$ ; and the pricing problem is:

$$\left\{ \begin{array}{l} \min \sum_{i=1}^n (c_i - \pi a_i) x_i - \pi A \\ \text{s. c. } \sum_{i=1}^n b_i x_i \leq B \\ x_i \in \{0, 1\}, i = 1, \dots, n. \end{array} \right.$$

The master problem associated to Lagrangian decomposition is given by:

$$\left\{ \begin{array}{l} \max \sum_{k=1}^K (\sum_{i=1}^n c_i x_i^{(k)}) \lambda_k \\ \sum_{k=1}^K (\sum_{i=1}^n x_i^{(k)}) \lambda_k - \sum_{l=1}^L (\sum_{i=1}^n y_i^{(l)}) \gamma_l = 0 \\ \sum_{k=1}^K \lambda_k = 1 \\ \sum_{l=1}^L \gamma_l = 1 \\ \lambda_k \geq 0, k = 1, \dots, K, \gamma_l \geq 0, l = 1, \dots, L \end{array} \right.$$

where  $x^{(k)}, k = 1, \dots, K$  (resp.  $y^{(l)}, l = 1, \dots, L$ ), are the extreme points of  $\text{Conv}(\{x_i \in \{0, 1\}, i = 1, \dots, n \mid \sum_{i=1}^n b_i x_i \leq B, \quad i = 1, \dots, n\})$  (resp.  $\text{Conv}(\{y_i \in \{0, 1\}, i = 1, \dots, n \mid \sum_{i=1}^n a_i y_i \leq A\})$ );

and the pricing problems are:

$$\begin{cases} \min & \sum_{i=1}^n u_i y_i \\ \text{s. c.} & \sum_{i=1}^n a_i y_i \leq A \\ & y_i \in \{0, 1\}, i = 1, \dots, n \end{cases}$$

and

$$\begin{cases} \min & \sum_{i=1}^n (c_i - u_i) x_i \\ \text{s. c.} & \sum_{i=1}^n b_i x_i \leq B \\ & x_i \in \{0, 1\}, i = 1, \dots, n. \end{cases}$$

where  $x_i$ ,  $i = 1, \dots, n$  and  $y_i$ ,  $i = 1, \dots, n$  are equal to 1 if object  $i$  is filled in the knapsack.

#### 4.2 The generalized assignment problem

It consists of finding a maximum profit assignment of  $T$  jobs to  $I$  agents such that each job is assigned to precisely one agent subject to capacity restrictions on the agents [10]. The standard integer programming formulation is the following:

$$\begin{cases} \max & \sum_i \sum_t c_{it} x_{it} \\ \text{s. c.} & \sum_i x_{it} = 1, \quad t = 1, \dots, T \\ & \sum_t r_{it} x_{it} \leq b_i, \quad i = 1, \dots, I \\ & x_{it} \in \{0, 1\}, \quad i = 1, \dots, I, t = 1, \dots, T \end{cases}$$

Two classical Dantzig-Wolfe decompositions can be made, by relaxing the assignment constraints or the capacity constraints.

The first classical Dantzig-Wolfe master problem is given by:

$$\begin{cases} \max & \sum_{k=1}^K (\sum_i \sum_t c_{it} x_{it}^{(k)}) \lambda_k \\ \text{s.c.} & \sum_{k=1}^K (\sum_i x_{it}^{(k)}) \lambda_k = 1, \quad t = 1, \dots, T \\ & \sum_{k=1}^K \lambda_k = 1 \\ & \lambda_k \geq 0, \quad k = 1, \dots, K \end{cases}$$

where  $x^{(k)}, k = 1, \dots, K$ , are the extreme points of  $Conv(\{x_{it} \in \{0, 1\} | \sum_t r_{it}x_{it} \leq b_i, i = 1, \dots, I\})$ ; and the associated pricing problem is:

$$\begin{cases} \min & \sum_i \sum_t (c_{it} - \pi_t)x_{it} - \sum_t \pi_t \\ \text{s. c.} & \sum_t r_{it}x_{it} \leq b_i, \quad i = 1, \dots, I \\ & x_{it} \in \{0, 1\}, i = 1, \dots, I, t = 1, \dots, T. \end{cases}$$

The second classical Dantzig-Wolfe master problem is given by:

$$\begin{cases} \max & \sum_{l=1}^L (\sum_i \sum_t c_{it}y_{it}^{(l)})\gamma_l \\ \text{s.c.} & \sum_{l=1}^L (\sum_t r_{it}y_{it}^{(l)})\gamma_l \leq b_i, \quad i = 1, \dots, I \\ & \sum_{l=1}^L \gamma_l = 1 \\ & \gamma_l \geq 0, \quad l = 1, \dots, L \end{cases}$$

where  $y^{(l)}, l = 1, \dots, L$  are the extreme points of  $Conv(\{y_{it} \in \{0, 1\} | \sum_i y_{it} = 1, t = 1, \dots, T\})$ ; and the associated pricing problem is:

$$\begin{cases} \min & \sum_i \sum_t (c_{it} - \pi_i)y_{it} - \sum_i \pi_i \\ \text{s. c.} & \sum_i y_{it} = 1, \quad t = 1, \dots, T \\ & y_{it} \in \{0, 1\}, i = 1, \dots, I, t = 1, \dots, T \end{cases}$$

The master problem associated to Lagrangian decomposition is given by:

$$\begin{cases} \max & \sum_{k=1}^K (\sum_i \sum_t c_{it}x_{it}^{(k)})\lambda_k \\ & \sum_{k=1}^K (\sum_i \sum_t x_{it}^{(k)})\lambda_k - \sum_{l=1}^L (\sum_i \sum_t y_{it}^{(l)})\gamma_l = 0 \\ & \sum_{k=1}^K \lambda_k = 1 \\ & \sum_{l=1}^L \gamma_l = 1 \\ & \lambda_k \geq 0, k = 1, \dots, K, \gamma_l \geq 0, l = 1, \dots, L \end{cases}$$

where  $x^{(k)}, k = 1, \dots, K$  (resp.  $y^{(l)}, l = 1, \dots, L$ ), are the extreme points of  $Conv(\{x_{it} \in \{0, 1\} | \sum_t r_{it}x_{it} \leq b_i, i = 1, \dots, I\})$  (resp.  $Conv(\{y_{it} \in \{0, 1\} | \sum_i y_{it} = 1, t = 1, \dots, T\})$ );

and the pricing problems are:

$$\begin{cases} \min \sum_i \sum_t u_{it} y_{it} \\ \text{s. c. } \sum_i y_{it} = 1, \quad t = 1, \dots, T \\ y_{it} \in \{0, 1\}, i = 1, \dots, I, t = 1, \dots, T \end{cases}$$

and

$$\begin{cases} \min \sum_i \sum_t (c_{it} - u_{it}) x_{it} \\ \text{s. c. } \sum_t r_{it} x_{it} \leq b_i, \quad i = 1, \dots, I \\ x_{it} \in \{0, 1\}, i = 1, \dots, I, t = 1, \dots, T. \end{cases}$$

where  $x_{it}$ ,  $i = 1, \dots, I$ ,  $t = 1, \dots, T$  and  $y_{it}$ ,  $i = 1, \dots, I$ ,  $t = 1, \dots, T$  are equal to 1 if job  $t$  is assigned to agent  $i$ .

## 5 Numerical experiments

This section is devoted to an experimental comparative study between LD and LR when solved by the CG algorithm. We consider the two optimization problems defined in the previous section : the 0-1 bidimensional knapsack problem and the generalized assignment problem.

We consider in our tests 6 instances of the 0-1 bi-dimensional knapsack problem from the OR-Library. Table 1 presents a comparative study between CG resolution of LD and LR formulations (denoted CG\_LD and CG\_LR respectively). The master and pricing problems are solved by CPLEX11.2 solver.

Table 1

Lagrangian relaxation and Lagrangian decomposition for (0-1 BKP)

	WEING1						WEING2					
	vR	%vE	Iter	tG	tSP	tMP	vR	%vE	Iter	tG	tSP	tMP
CG_LR	141 388.50	0.1	6	0.12	0.12	0.00	130 883.00	0.0	1	0.01	0.01	0.00
CG_LD	141 383.00	0.1	136	9.55	8.72	0.24	130 883.00	0.0	157	13.61	12.56	0.40
	WEING3						WEING4					
	vR	%vE	Iter	tG	tSP	tMP	vR	%vE	Iter	tG	tSP	tMP
CG_LR	97 613.92	2.0	5	0.13	0.11	0.00	122 321.58	2.5	7	0.08	0.06	0.01
CG_LD	95 677.00	0.0	142	11.42	10.64	0.25	119 337.00	0.0	156	12.68	11.54	0.33
	WEING5						WEING6					
	vR	%vE	Iter	tG	tSP	tMP	vR	%vE	Iter	tG	tSP	tMP
CG_LR	98 796.00	0.0	1	0.01	0.00	0.01	130 697.80	0.1	6	0.05	0.05	0.00
CG_LD	98 796.00	0.0	77	3.51	2.99	0.16	130 623.00	0.0	162	12.47	11.51	0.33

vR: the relaxation value.  
 %vE: the gap between relaxation and optimal values.  
 Iter: number of iterations.  
 tG: the global resolution time (s).  
 tSP: the global resolution time of pricing problems (s).  
 tM: cumulated master problems resolution time (s).

CG\_LR and CG\_LD optimality are reached for all instances. As expected, LD gives better upper bounds than LR. On average on instances WEING $i$ ,  $i = 1, \dots, 6$ , %vE associated to LD (resp. RL) is 0.02 (resp. 0.78), but we observe that the average resolution time of CG\_LR (0.07 s) is very small compared to CG\_LD computation time (10.54 s), this is due to the fact that the computational effort of each CG\_LD iteration is greater than the CG\_LR one and to the slow convergence of CG\_LD compared to CG\_LR.

We consider also in our tests 6 instances of the GAP from the OR-Library. All instances gap $i$ ,  $i = 1, \dots, 6$  have the same size, 5 agents and 15 jobs. The master and pricing problems are solved by CPLEX11.2 solver. Table 2 shows a comparison between LR and LD algorithms performances, when we apply for LR the second classical Dantzig-Wolfe decomposition, by relaxing the capacity constraints (cf section 4.2).

Table 2  
Lagrangian relaxation and Lagrangian decomposition for (GAP)

	gap1						gap2					
	vR	%vE	Iter	tG	tSP	tMP	vR	%vE	Iter	tG	tSP	tMP
CG_LR	343.59	2.3	33	0.27	0.16	0.03	339.38	3.8	26	0.22	0.17	0.00
CG_LD	337.00	0,3	1169	383.13	343.61	29.37	327.00	0,0	894	258.41	234.55	15.78
	gap3						gap4					
	vR	%vE	Iter	tG	tSP	tMP	vR	%vE	Iter	tG	tSP	tMP
CG_LR	349.68	3.2	33	0.22	0.14	0.01	350.40	2.8	31	0.25	0.17	0.00
CG_LD	339.50	0.1	945	273.18	245.89	19.01	341.00	0.0	878	282.25	258.89	15.74
	gap5						gap6					
	vR	%vE	Iter	tG	tSP	tMP	vR	%vE	Iter	tG	tSP	tMP
CG_LR	335.76	3.0	35	0.28	0.17	0.05	351.82	2.0	30	0.22	0.12	0.08
CG_LD	327.25	0.4	595	163.86	149.73	9.05	345.00	0.0	1115	334.65	301.99	23.93

vR: the relaxation value.  
 %vE: the gap between relaxation and optimal values.  
 Iter: number of iterations.  
 tG: the global resolution time (s).  
 tSP: the global resolution time of pricing problems (s).  
 tMP: cumulated master problems resolution time (s).

As before, CG\_LR and CG\_LD optimality are reached for all instances. LD gives better upper bounds than LR. On average on instances gap $i$ ,  $i = 1, \dots, 6$ , %vE associated to LD (resp. RL) is 0.13 (resp. 2.85), but we observe that the average resolution time of CG\_LR (0.24 s) is still very small compared to CG\_LD computation time (282.58 s).

The first classical Dantzig-Wolfe decomposition for LR, by relaxing the assignment constraints (cf section 4.2), has been also tested on the same instances, the results show that the bounds are tighter (but they are not better than those obtained by LD) and the column generation algorithm takes more iterations and time to converge.

## 6 Conclusion

This paper focused on Dantzig-Wolfe Decomposition principle. We propose a new Dantzig-Wolfe master problem for integer linear programming, which allows to propose an alternative dominance proof of Lagrangian Decomposition bound over Lagrangian Relaxation bound. As illustration, we have given the two Dantzig-Wolfe decomposition models for the 0-1 Bi-dimensional Knapsack Problem and the Generalized Assignment Problem. The obtained experimental results demonstrate the superiority of the Lagrangian Decomposition bound, but the gain on bound quality impose an additional computation effort. In fact, at each iteration of the column generation algorithm for the Lagrangian decomposition, two pricing problems (generally integer problems) have to be solved. Through this experimental study, we conclude that column generation resolution of Lagrangian decomposition can be useful if we want to obtain a good initial bound, as for example at the root node of a branch and bound or a branch and price scheme.

## References

- [1] GB. Dantzig and P. Wolfe, Decomposition principle for linear programs, *Operations Research*, 8:101-111, 1960.
- [2] A.M. Geoffrion, Lagrangian relaxation for integer programming, *Mathematical Programming Stud.*, 2:82-114, 1974.
- [3] J. Desrosiers and Y. Dumas and M. M. Solomon and F. Soumis, Time Constrained Routing et Scheduling, In *Handbooks in Operations Research and Management Science*, Vol 8: Network Routing. M. O. Ball, T. L. Magnanti, and G. L. Nemhauser (eds). North-Holland, Amsterdam, The Netherlands., 1995.
- [4] F. Glover, A multiphase dual algorithm for the 0-1 integer programming problem, *Operations Research*, 13(6):879-919, 1965.
- [5] M. Guignard and S. Kim, Lagrangian decomposition: a model yielding stronger langrangian bounds, *Mathematical programming*, 32:215-228, 1987.
- [6] M. Guignard and S. Kim, Lagrangian decomposition for integer programming: theory and applications, *R.A.I.R.O*, 21:307-324, 1987.
- [7] J.E. Kelley, The cutting-plane method for solving convex programs, *SIAM Journal on Optimization*, 8:703-712, 1960.
- [8] L.S. Lasdon, *Optimization theory for large systems*, Macmillan series in operations research, 1972.
- [9] C. Lemaréchal, The omnipresence of Lagrange, *4OR*, 1(1) :725, 2003.

- [10] S. Martello and P.Toth, Generalized assignment problems, Lecture Notes, In Computer Science 660:351-369, 1992.
- [11] P. Michelon, Méthodes lagrangiennes pour la programmation linéaire avec variables entières, Investigación Operativa, 2(2):127-146, 1991.
- [12] A. Nagih, G. Plateau, A Lagrangian Decomposition for 0-1 Hyperbolic Programming Problems, International Journal of Mathematical Algorithms, 14:299-314, 2000.
- [13] A. Nagih, G. Plateau, Dualité lagrangienne en programmation fractionnaire concave-convexe en variables 0-1, CRAS : Comptes Rendus de l'Académie des Sciences de Paris, tome 331, série I, pp. 491-496, 2000.
- [14] H. Reinoso and N. Maculan, Lagrangian decomposition for integer programming: a new scheme, INFOR, 52(2):1-5, 1992.